



US Patent & Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

+constant +pool +entry +unresolved +value +field



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **constant pool entry unresolved value field**Found **34** of 148,786

Sort results by

relevance

Display results

expanded form

[Save results to a Binder](#) [Search Tips](#)☐ Open results in a new window[Try an Advanced Search](#)[Try this search in The ACM Guide](#)

Results 1 - 20 of 34

Result page: **1** [2](#) [next](#)Relevance scale ☐ ☐ ☐ ☐ ☐**1** [Dynamic speculation and synchronization of data dependences](#)

Andreas Moshovos, Scott E. Breach, T. N. Vijaykumar, Gurindar S. Sohi

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2Full text available: [pdf\(2.61 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Data dependence speculation is used in instruction-level parallel (ILP) processors to allow early execution of an instruction before a logically preceding instruction on which it may be data dependent. If the instruction is independent, data dependence speculation succeeds; if not, it fails, and the two instructions must be synchronized. The modern dynamically scheduled processors that use data dependence speculation do so blindly (i.e., every load instruction with unresolved dependences is spec ...

2 [A framework for efficient reuse of binary code in Java](#)

Pramod G. Joisha, Samuel P. Midkiff, Mauricio J. Serrano, Manish Gupta

June 2001 **Proceedings of the 15th international conference on Supercomputing**Full text available: [pdf\(419.49 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper presents a compilation framework that enables efficient sharing of executable code across distinct Java Virtual Machine (JVM) instances. High-performance JVMs rely on run-time compilation, since static compilation cannot handle many dynamic features of Java. These JVMs suffer from large memory footprints and high startup costs, which are serious problems for embedded devices (such as hand held personal digital assistants and cellular phones) and scalable servers. A recently propose ...

3 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**Full text available: [pdf\(4.21 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

4 [Information systems outsourcing: a survey and analysis of the literature](#)

Jens Dibbern, Tim Goles, Rudy Hirschheim, Bandula Jayatilaka

November 2004 **ACM SIGMIS Database**, Volume 35 Issue 4Full text available: [pdf\(1.51 M...\)](#)Additional Information: [full citation](#), [abstract](#), [references](#)


In the last fifteen years, academic research on information systems (IS) outsourcing has evolved rapidly. Indeed the field of outsourcing research has grown so fast that there has been scant opportunity for the research community to take a collective breath, and complete a global assessment of research activities to date. This paper seeks to address this need by exploring and synthesizing the academic literature on IS outsourcing. It offers a roadmap of the IS outsourcing literature, highlight ...

Keywords: determinants, literature review, outcomes, outsourcing, relationships, research approaches, theoretical foundations

5 Query evaluation techniques for large databases

Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

Full text available:  [pdf\(9.37 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

6 A first-class approach to genericity

Eric Allen, Jonathan Bannet, Robert Cartwright

October 2003 **ACM SIGPLAN Notices , Proceedings of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 38 Issue 11

Full text available:  [pdf\(357.33 KB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper describes how to add first-class generic types---including mixins---to strongly-typed OO languages with nominal subtyping such as Java and C#. A generic type system is "first-class" if generic types can appear in any context where conventional types can appear. In this context, a mixin is simply a generic class that extends one of its type parameters, e.g., a class $C<T>$ that extends T . Although mixins of this form are widely used in Cpp (via templates), they are clumsy an ...

7 The treatment of data types in EL1

Ben Wegbreit

May 1974 **Communications of the ACM**, Volume 17 Issue 5

Full text available:  [pdf\(1.44 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In constructing a general purpose programming language, a key issue is providing a sufficient set of data types and associated operations in a manner that permits both natural problem-oriented notation and efficient implementation. The EL1-language contains a number of features specifically designed to simultaneously satisfy both requirements. The resulting treatment of data types includes provision for programmer-defined data types and generic routines, programmer control over type conver ...

Keywords: coercion, compilation, data, data type definition, data types, description language, extensible languages, generic functions, mode unions, modes, type conversion

8 Computing curricula 2001

September 2001 **Journal on Educational Resources in Computing (JERIC)**Full text available:  [pdf\(613.63 KB\)](#)
 [html\(2.78 KB\)](#)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)**9** [A type-checking program linkage system for pascal](#)


R. B. Kieburtz, W. Barabash, C. R. Hill

May 1978 **Proceedings of the 3rd international conference on Software engineering**Full text available:  [pdf\(618.46 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We describe a new software facility useful in the development, debugging, and integration of quasi-independent program modules. It has been implemented in conjunction with a Pascal compiler for IBM 360 computers, but the linkage facility is machine-independent up to the point of formatting system object modules. With some minor syntactic extensions to permit the designation of external references, Pascal becomes a powerful language for modular programming. The new linkage subsystem ...

10 [Adoption, diffusion, and infusion of IT: Organizational adoption and assimilation of complex technological innovations: development and application of a new framework](#)

Michael J. Gallivan

July 2001 **ACM SIGMIS Database**, Volume 32 Issue 3Full text available:  [pdf\(3.37 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper explores the applicability of traditional innovation adoption and diffusion models to contingent, authority innovation processes occurring within an organizational context (Zaltman, Duncan & Holbeck, 1973); that is, when employees in organizations adopt an innovation that has been chosen by an authority figure. This paper identifies existing gaps in traditional innovation adoption models and concludes that a new framework is required -- one that incorporates the unique processes and ...

Keywords: technology adoption, technology diffusion**11** [The SimpleScalar tool set, version 2.0](#)

Doug Burger, Todd M. Austin

June 1997 **ACM SIGARCH Computer Architecture News**, Volume 25 Issue 3Full text available:  [pdf\(985.46 KB\)](#)Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This document describes release 2.0 of the SimpleScalar tool set, a suite of free, publicly available simulation tools that offer both detailed and high-performance simulation of modern microprocessors. The new release offers more tools and capabilities, precompiled binaries, cleaner interfaces, better documentation, easier installation, improved portability, and higher performance. This paper contains a complete description of the tool set, including retrieval and installation instructions, a d ...

12 [Implementing optimizations at decode time](#)

Ilhyun Kim, Mikko H. Lipasti

May 2002 **ACM SIGARCH Computer Architecture News**, Volume 30 Issue 2Full text available:  [pdf\(1.26 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)[Publisher Site](#)

The number of pipeline stages separating dynamic instruction scheduling from instruction execution has increased considerably in recent out-of-order microprocessor implementations, forcing the scheduler to allocate functional units and other execution resources several cycles before they are actually used. Unfortunately, several proposed microarchitectural optimizations become less desirable or even impossible in such an environment, since they require instantaneous or near-instantaneous changes ...

Keywords: speculative scheduling, speculative decode, runtime optimizations, silent store, reference combining, confidence prediction

13 TRIPS: A polymorphous architecture for exploiting ILP, TLP, and DLP

Karthikeyan Sankaralingam, Ramadass Nagarajan, Haiming Liu, Changkyu Kim, Jaehyuk Huh, Nitya Ranganathan, Doug Burger, Stephen W. Keckler, Robert G. McDonald, Charles R. Moore
March 2004 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume 1

Issue 1

Full text available:  [pdf\(832.30 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper describes the *polymorphous* TRIPS architecture that can be configured for different granularities and types of parallelism. The TRIPS architecture is the first in a class of post-RISC, dataflow-like instruction sets called explicit data-graph execution (EDGE). This EDGE ISA is coupled with hardware mechanisms that enable the processing cores and the on-chip memory system to be configured and combined in different modes for instruction, data, or thread-level parallelism. To adapt ...

Keywords: Computer architecture, configurable computing, scalable and high-performance computing

14 Instruction architecture of an aerospace multiprocessor

James S. Miller, Woodrow H. Vandever

November 1973 **ACM SIGPLAN Notices , Proceedings of a symposium on High-level-language computer architecture**, Volume 8 Issue 11

Full text available:  [pdf\(592.78 KB\)](#) Additional Information: [full citation](#), [references](#)

15 Instruction architecture of an aerospace multiprocessor

James S. Miller, Woodrow H. Vandever

November 1973 **Proceedings of the ACM-IEEE symposium on High-level-language computer architecture**

Full text available:  [pdf\(598.38 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper describes the architecture, the data forms, and the instruction set of a multiprocessor computer designed to provide the central computational facilities of a long-lifetime orbiting space laboratory. An overview of the system [M73] describes other aspects of the design, including its high level of fault-tolerance, flexible capacity, real time control capability, and process security in the presence of software faults. The present treatment emphasizes the considerations which led ...

16 Ada development system technical and performance requirements (with rationale)

Donald G. Krantz

December 1990 **Proceedings of the conference on TRI-ADA '90**

Full text available:  [pdf\(1.85 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

This paper discusses requirements for Ada1 compilers and associated tools used for real-time embedded weapons systems (EWS) development. The requirements have been developed over a period of several years by embedded systems developers at Honeywell Inc. and Alliant Techsystems Inc. Requirements for the run time system, compiler-generated code, and host tools such as linkers are presented. A short rationale statement is provided with each specific requirement.

17 Efficient binary I/O of IDL objects

J. M. Newcomer

November 1987 **ACM SIGPLAN Notices**, Volume 22 Issue 11

Full text available:  [pdf\(916.20 KB\)](#) Additional Information: [full citation](#), [citations](#), [index terms](#)

18 Technical contributions: Language design for the Ironman requirement: notes and examples

Mary Shaw, Paul Hilfinger, Wm. A. Wulf

September 1978 **ACM SIGPLAN Notices**, Volume 13 Issue 9

Full text available:  [pdf\(900.37 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

The Tartan language was designed as an experiment to see whether the Ironman requirement for a common high-order programming language could be satisfied by an extremely simple language. The result, Tartan substantially meets the Ironman requirement. We believe it is substantially simpler than the four designs that were done in the first phase of the DOD-1 effort. The language definition appears in a companion report; this report provides a more expository discussion of some of the language's features ...

19 Centralized versus decentralized computing: organizational considerations and management options

John Leslie King


December 1983 **ACM Computing Surveys (CSUR)**, Volume 15 Issue 4

Full text available:  [pdf\(3.09 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

20 Pragmatic techniques for program analysis and verification

Erhard Ploedereder

September 1979 **Proceedings of the 4th international conference on Software engineering**

Full text available:  [pdf\(860.52 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The Program Development System (PDS) is a collection of programming tools created as an extension of the ECL programming system²³. It contains components that assist the programmer in the definition and modular structuring of large programs at different levels of algorithmic abstraction. These components are supplemented by a program analysis package that produces an information pool to be used for such tasks as source-to-source optimization, semi-automated program documentation ...

Results 1 - 20 of 34

Result page: [1](#) [2](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)



US Patent & Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **constant pool entry unresolved value field**Found **34** of **148,786**

Sort results by

Display results

☒ [Save results to a Binder](#)
☒ [Search Tips](#)
☐ [Open results in a new window](#)
[Try an Advanced Search](#)
[Try this search in The ACM Guide](#)

Results 21 - 34 of 34

Result page: [previous](#) [1](#) [2](#)Relevance scale ☐ ☐ ☐ ☐ ☐**21** [Principles and realization strategies of multilevel transaction management](#)

Gerhard Weikum

March 1991 **ACM Transactions on Database Systems (TODS)**, Volume 16 Issue 1Full text available: [pdf\(3.72 MB\)](#)
 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

One of the demands of database system transaction management is to achieve a high degree of concurrency by taking into consideration the semantics of high-level operations. On the other hand, the implementation of such operations must pay attention to conflicts on the storage representation levels below. To meet these requirements in a layered architecture, we propose a multilevel transaction management utilizing layer-specific semantics. Based on the theoretical notion of multilevel serial ...

Keywords: atomicity persistence concurrency control, multilevel transactions, persistence, serializability

22 [Smart recompilation](#)

Walter F. Tichy

June 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 3Full text available: [pdf\(1.56 MB\)](#)
 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

With current compiler technology, changing a single line in a large software system may trigger massive recompilations. If the change occurs in a file with shared declarations, all compilation units depending upon that file must be recompiled to assure consistency. However, many of those recompilations may be redundant, because the change may affect only a small fraction of the overall system. Smart recompilation is a method for reducing the set of modules that must be recompiled ...

23 [Compression & aggregation: Model-based compression in wireless ad hoc networks](#)

Milenko Drinic, Darko Kirovski, Miodrag Potkonjak


November 2003 **Proceedings of the 1st international conference on Embedded networked sensor systems**Full text available: [pdf\(347.48 KB\)](#)
 Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We present a technique for compression of shortest paths routing tables for wireless ad hoc networks. The main characteristic of such networks is that geographic location of nodes determines network topology. As opposed to encoding individual node locations, at each node our approach groups the remaining nodes in the network into regions. All shortest paths to nodes in a specific region are routed via the same neighboring node. In this paper, we propose an algorithm for dividing a network field ...

Keywords: compression, modeling, routing protocols, routing tables, sensor networks, trajectory

24 RAID: high-performance, reliable secondary storage

Peter M. Chen, Edward K. Lee, Garth A. Gibson, Randy H. Katz, David A. Patterson
June 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 2

Full text available:  pdf(3.60 MB)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Disk arrays were proposed in the 1980s as a way to use parallelism between multiple disks to improve aggregate I/O performance. Today they appear in the product lines of most major computer manufacturers. This article gives a comprehensive overview of disk arrays and provides a framework in which to organize current and future work. First, the article introduces disk technology and reviews the driving forces that have popularized disk arrays: performance and reliability. It discusses the tw ...

Keywords: RAID, disk array, parallel I/O, redundancy, storage, striping

25 High-speed switch scheduling for local-area networks

Thomas E. Anderson, Susan S. Owicki, James B. Saxe, Charles P. Thacker
November 1993 **ACM Transactions on Computer Systems (TOCS)**, Volume 11 Issue 4

Full text available:  pdf(2.37 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Current technology trends make it possible to build communication networks that can support high-performance distributed computing. This paper describes issues in the design of a prototype switch for an arbitrary topology point-to-point network with link speeds of up to 1 Gbit/s. The switch deals in fixed-length ATM-style cells, which it can process at a rate of 37 million cells per second. It provides high bandwidth and low latency for datagram traffic. In addition, it supports real-time t ...

Keywords: ATM networks, iterative matching, statistical matching, switching scheduling

26 A simulation model for network routing

Udo W. Pooch, Charles Neblock, Rahul Chattergy
December 1979 **Proceedings of the 11th conference on Winter simulation - Volume 2**

Full text available:  pdf(970.64 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The simulation program described in this paper was devised as a vehicle for the study of communication network routing procedures. It was designed to model the behavior of a wide range of network topologies and routing disciplines. An event driven simulation approach was chosen to minimize program development time and complexity. GASP-IV was selected as the simulation language. The determining factor in choosing the language was the clean interface it presents to FORTRAN. The use ...

27 Reprints: Becoming a computer scientist

Amy Pearl, Martha E. Pollack, Eve Riskin, Becky Thomas, Elizabeth Wolf, Alice Wu
June 2002 **ACM SIGCSE Bulletin**, Volume 34 Issue 2

Full text available:  pdf(1.18 MB)

Additional Information: [full citation](#), [references](#), [citations](#)


28 A C++ data model supporting reachability analysis and dead code detection

Yih-Farn R. Chen, Emden R. Gansner, Eleftherios Koutsofios
November 1997 **ACM SIGSOFT Software Engineering Notes, Proceedings of the 6th European conference held jointly with the 5th ACM SIGSOFT international symposium on Foundations of software engineering**, Volume 22 Issue 6

Full text available:  [pdf\(1.33 MB\)](#)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

29 An interview with the 1986 A. M. Turing Award recipients—John E. Hopcroft and Robert E. Tarjan

Karen A. Frenkel

March 1987 **Communications of the ACM**, Volume 30 Issue 3Full text available:  [pdf\(1.39 MB\)](#)Additional Information: [full citation](#), [abstract](#), [index terms](#)

In the following interview, which took place at the 1986 Fall Joint Computer Conference in Dallas, Texas, John Hopcroft and Robert Tarjan discuss their collaboration and its influence on their separate research today. They also comment on supercomputing and parallelism, particularly with regard to statements by FJCC Keynote speakers Kenneth Wilson, Nobel laureate and director of Cornell University's Supercomputer Center, and C. Gordon Bell, chief architect on the team that designed DEC's VA ...

30 High speed switch scheduling for local area networks

Thomas E. Anderson, Susan S. Owicki, James B. Saxe, Charles P. Thacker

September 1992 **ACM SIGPLAN Notices, Proceedings of the fifth international conference on Architectural support for programming languages and operating systems**, Volume 27 Issue 9Full text available:  [pdf\(1.25 MB\)](#)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

31 OceanStore: an architecture for global-scale persistent storage


John Kubiawicz, David Bindel, Yan Chen, Steven Czerwinski, Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Chris Wells, Ben Zhao

November 2000 **Proceedings of the ninth international conference on Architectural support for programming languages and operating systems**, Volume 28, 34 Issue 5, 5Full text available:  [pdf\(166.53 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

OceanStore is a utility infrastructure designed to span the globe and provide continuous access to persistent information. Since this infrastructure is comprised of untrusted servers, data is protected through redundancy and cryptographic techniques. To improve performance, data is allowed to be cached anywhere, anytime. Additionally, monitoring of usage patterns allows adaptation to regional outages and denial of service attacks; monitoring also enhances performance through pro-active movement ...


32 OceanStore: an architecture for global-scale persistent storage

John Kubiawicz, David Bindel, Yan Chen, Steven Czerwinski, Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Westley Weimer, Chris Wells, Ben Zhao

November 2000 **ACM SIGPLAN Notices**, Volume 35 Issue 11Full text available:  [pdf\(1.47 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


OceanStore is a utility infrastructure designed to span the globe and provide continuous access to persistent information. Since this infrastructure is comprised of untrusted servers, data is protected through redundancy and cryptographic techniques. To improve performance, data is allowed to be cached anywhere, anytime. Additionally, monitoring of usage patterns allows adaptation to regional outages and denial of service attacks; monitoring also enhances performance through pro-active movement ...

33 Workshop on object-oriented programming ECOOP 1987. Paris, June 18, 1987

January 1988 **ACM SIGPLAN Notices**, Volume 23 Issue 1Full text available:  [pdf\(1.22 MB\)](#)Additional Information: [full citation](#), [citations](#), [index terms](#)

34 Proof linking: modular verification of mobile programs in the presence of lazy, dynamic linking

Philip W. L. Fong, Robert D. Cameron

October 2000 **ACM Transactions on Software Engineering and Methodology (TOSEM)**,
Volume 9 Issue 4Full text available:  pdf(233.60 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#),
[review](#)



Although mobile code systems typically employ link-time code verifiers to protect host computers from potentially malicious code, implementation flaws in the verifiers may still leave the host system vulnerable to attack. Compounding the inherent complexity of the verification algorithms themselves, the need to support lazy, dynamic linking in mobile code systems typically leads to architectures that exhibit strong interdependencies between the loader, the verifier, and the linker. To simp ...

Keywords: Java, correctness conditions, dynamic linking, mobile code, modularity, proof linking, safety, verification protocol, virtual machine architecture

Results 21 - 34 of 34

Result page: [previous](#) [1](#) [2](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)